Constructing Workflows by Integrating Interactive Information Sources

Amarnath Gupta

Ilkay Altintas

Bertram Ludäscher

Reagan W. Moore

San Diego Supercomputer Center, UCSD

Related Projects

- Integration of Neuroscience Information for things like:
  - Mouse models of human disease
  - Protein localization
  - Comparative gene expression over embryonic development
- Integrative construction and analysis of Yeast Gene Regulatory Network for sporulation/meiosis

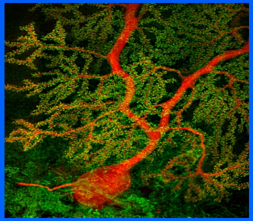
A Neuroscientist’s Information Integration Problem

What is the cerebellar distribution of rat proteins with more than 70% homology with human NCS-1? Any structure specificity? How about other rodents?

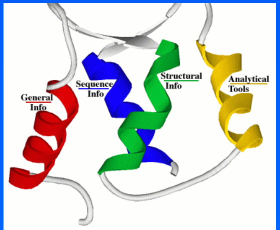
?

Information Integration

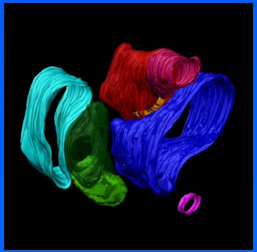
“Complex Multiple-Worlds” Mediation



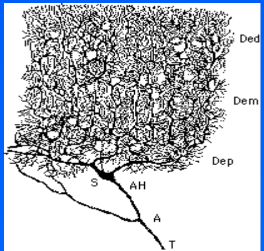
protein localization
(NCMIR)



sequence info
(CaPROT)



morphometry
(SYNAPSE)



neurotransmission
(SENSELAB)

But this work looks at a *different* information integration problem

Let’s first see a Demo of the Application Scenario

Was that good enough?

Why can’t we just use Discovery Links from IBM to do this?

An Equivalent Query for the Task

Assumin **uage**

Find all human transcription factors that bind to promoter regions of those genes that hybridize well (top 3) with my sample cDNA or its 3 closest homologues. Report the genes, the homologues and the transcription factors

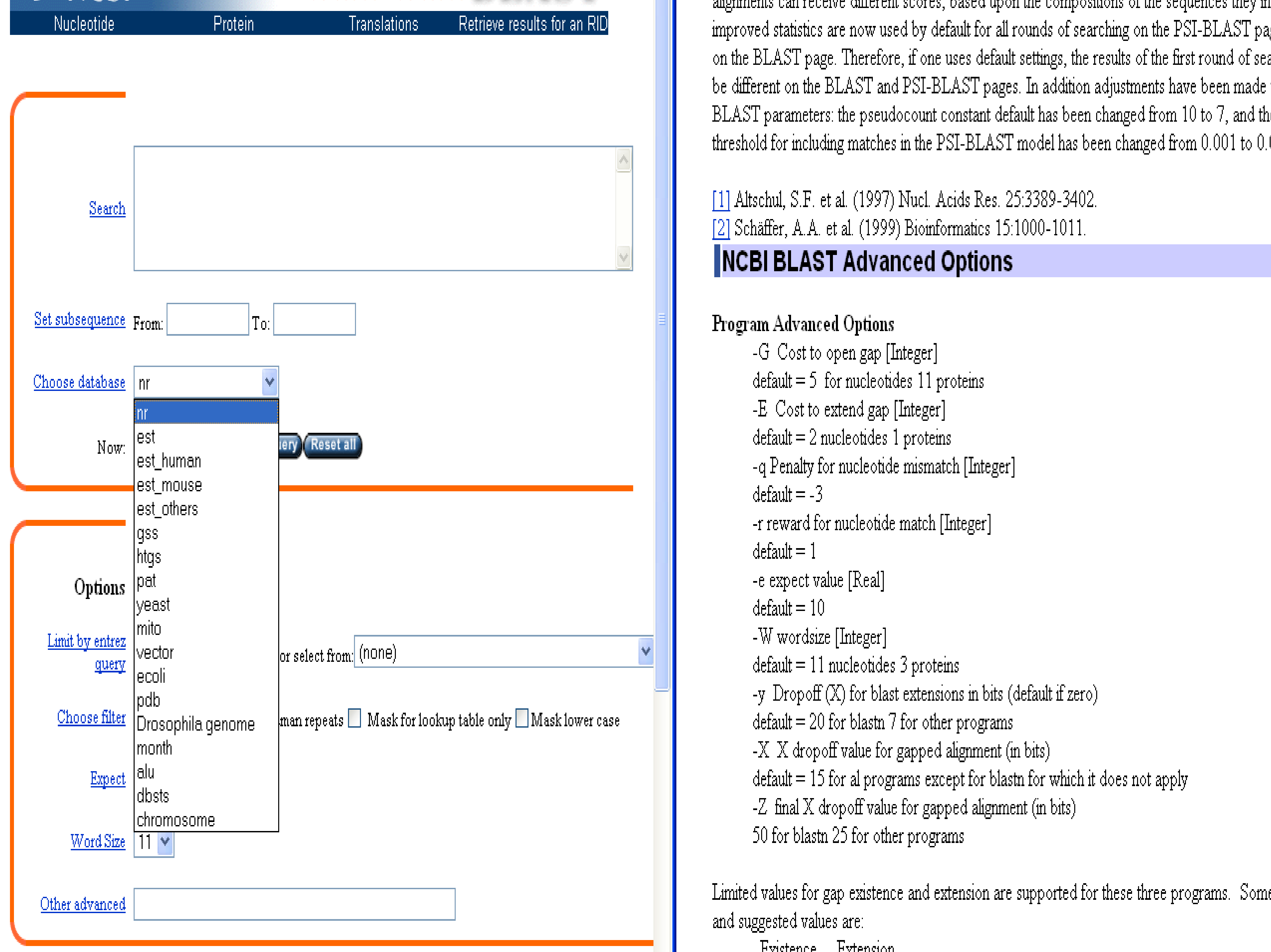
```
select      gene g, homol, t_factor tf
from        clusfavor C, genebank G, sample S,
            ncbi NC, transfac T, matinspector M
where       g in top(3, C.rank_by_CV(S)) and
            gs is G.sequence(g) and
            homol in top(3, NC.blast_search(gs, …)) and
            prom is extend_limits(homol) and
            tf in M.get_tfs(prom, core_sim, matrix_sim,
                               'vertebrate_matrix') and
            'human' in T.species_of(tf)
```

The evaluation plan for this query would be very close to our "workflow"

Why not take a "pure" mediation approach to the problem?

- Some essential facts about a mediator system
  - A traditional mediator can execute a single query plan
  - A mediator with an adaptive query plan generator can perform *mid-stream plan corrections* based on properties like
    - source availability
    - data rate
    - size of intermediate results
  - Semantic dependencies between data from multiple sources are handled statically at the time of view definition but not during query execution
- Mapping that to our problem …

  is very difficult … here is why



alignments can receive different scores, based upon the compositions of the sequences they m improved statistics are now used by default for all rounds of searching on the PSI-BLAST pa on the BLAST page. Therefore, if one uses default settings, the results of the first round of sea be different on the BLAST and PSI-BLAST pages. In addition adjustments have been made BLAST parameters: the pseudocount constant default has been changed from 10 to 7, and th threshold for including matches in the PSI-BLAST model has been changed from 0.001 to 0.0

[1] Altschul, S.F. et al. (1997) Nucl. Acids Res. 25:3389-3402.
[2] Schäffer, A.A. et al. (1999) Bioinformatics 15:1000-1011.

NCBI BLAST Advanced Options

Program Advanced Options

-G Cost to open gap [Integer]
default = 5 for nucleotides 11 proteins
-E Cost to extend gap [Integer]
default = 2 nucleotides 1 proteins
-q Penalty for nucleotide mismatch [Integer]
default = -3
-r reward for nucleotide match [Integer]
default = 1
-e expect value [Real]
default = 10
-W wordsize [Integer]
default = 11 nucleotides 3 proteins
-y Dropoff (X) for blast extensions in bits (default if zero)
default = 20 for blastn 7 for other programs
-X X dropoff value for gapped alignment (in bits)
default = 15 for al programs except for blastn for which it does not apply
-Z final X dropoff value for gapped alignment (in bits)
50 for blastn 25 for other programs

Limited values for gap existence and extension are supported for these three programs. Some and suggested values are:

Existence Extension

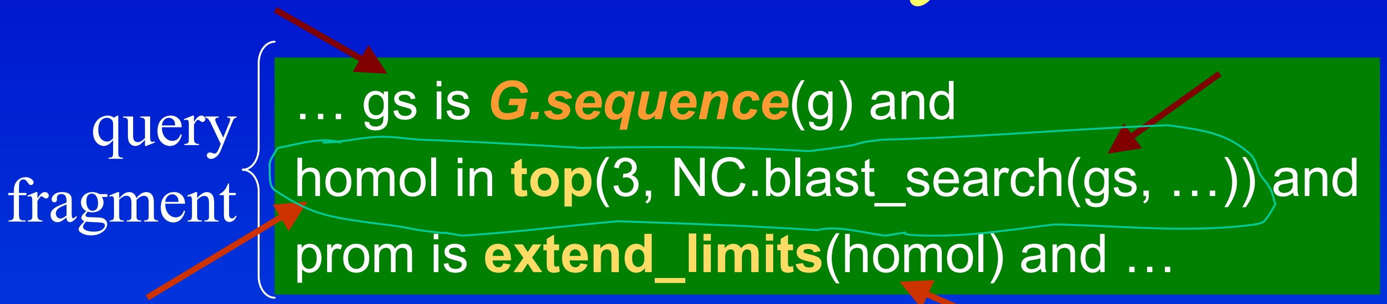
Modeling an Interactive Source for Integration

- Modeling the clicking/form-filling mechanics
  - Single page
    - Queries with binding patterns
  - Multi-page
    - Correlated queries with implicit joins or passing of fixed parameters between them
    - Management of intermediate variables
  - A source is *wrappable* if all the operations on it can be expressed as parameterized PSJ queries over the set of pages

Modeling an Interactive Source for Integration

- Modeling ***Interaction Semantics***
  - How are the *query parameters* **constrained by** the attributes of the input *data objects*?
  - How does the *parameter adjustment process* **depend on** the properties of the *intermediate data*?
  - How do we know when an iteration **terminates**?
  - When can we **exit a source** to go to the next one?
  - When do we need to **return** to the current source?
  - Which variables does the system need to keep for
    - interacting with the next source?
    - returning to the same source?
- What in this ***can*** be automated?

Control-Extensibility in Mediators



query fragment

… gs is ***G.sequence***(g) and
homol in **top**(3, NC.blast_search(gs, …)) and
prom is **extend_limits**(homol) and …

- Rule 1: *if gs is a complete* ***known*** *gene sequence then convert gs into equivalent protein and then perform protein_blast else perform a nucleotide_blast*
- Technique 2:

```
repeat{
                  results:=blast_search(…);
                  if(test_quality(top(3, results))= ok) {
                            report homol:= top(3, results)
                            exit_local;
                  }
                  else {…}
         } until test_converge(results);
```

- Rule 3:

```
case species(homol) of{
          bakers_yeast: extension = 1000;
          c_elegans: extension = 3000;
          drosophila: extension = eval(wrapper(homol, http://www.drosopila.org/, …);
   …
      }
```

Conclusions
(for now)

- The problem requirements do not fit
  - Current query decomposition/rewrite models
  - Traditional workflow models
- Next Tasks
  - Get a ***BETTER FUNCTIONAL SPEC***
  - Formal Extension of Query Capabilities with Interaction Semantics
  - Develop an operational API for interaction specification
  - Create a query rewriting method partial execution-control fragments, possibly by plugging-in user-defined control structures
- A Not-so-far-term task
  - Connect this to the Storage Resource Broker and the Teragrid facilities